7. Manuel

Program Name: Manuel.java **Input File: manuel.dat**

In Manuel's Algebra II class, he has just learned about solving two equations with two unknowns. For example, if Manuel is given:

$$2x - 1y = -4$$
$$1x + 3y = 5$$

Manuel knows the solution to be $x = -1$ and $y = 2$. Manuel was taught there is more than one way to arrive at the solution. Examples include elimination, substitution, and even some slick matrix operations to derive this solution. Manuel doesn't have too much trouble with the two equations and two unknown problems, but every now and then, Manuel's teacher gives him three equations and three unknowns. For example:

$$1x + 3y - 1z = 2$$
$$4x + 2y + 5z = 1$$
$$3x + 0y + 1z = 12$$

Manuel knows the solution to be $x = 5$, $y = 3$, and $z = -2$. Manuel has the idea to write a Java program that can solve either two equations with two unknowns problems, or three equations with three unknowns problems. Do you think you can assist Manuel with this problem?

Input: Input will consist of a single integer T, the number of test cases. T will be in range of [1,20]. Each test case will then begin with an integer V, either a 2 or a 3 on the first line. V signifies where the test case has either two equations unknowns, or three equations and three unknowns. The following V lines will consist of the equations. Equations with two unknowns will have the form: `Avar1+Bvar2=D` where `A,B,` and `D` are guaranteed to be integers from `[-2147483648,-2147483647]` and `var1,` and `var2` are guaranteed to be a single, lowercase letter from `'a' - 'z'`. Variables are not allowed to be the same and are not allowed to be duplicated within the same test case. Equations with three unknowns will have the form: `Avar1+Bvar2+Cvar3=D` where `A,B, C,` and `D` are guaranteed to be integers from `[-2147483648,-2147483647]` and `var1, var2,` and `var3` are guaranteed to be a single, lowercase letter from `'a' - 'z'`. Variables will appear in the same order for each of the equations in the test case. Variables are not allowed to be the same and are not allowed to be duplicated within the same test case.

Output: For each test case you are to output on a single line: `var1=NUM1,var2=NUM2,var3=NUM3` where `var1, var2,` and `var3` are the variables read in as input (see above), and `NUM1, NUM2,` and `NUM3` are floating point values rounded to three decimal places. An output of zero must be postive. `NUM1, NUM2,` and `NUM3` when input into the original equations, should satisfy the equations. The order of `var1, var2,` and `var3` should match the order as given in the equations. It can also be guaranteed that there is one and only one solution to the equations.

~Sample Input and Output on next page~

~Manuel continued~

Sample input:

```
7
3
1x+1y+1z=6
0x+2y+5z=-4
2x+5y-1z=27
2
3t+1a=-10
1t-2a=4
3
-4x-5y-1z=18
-2x-5y-2z=12
-2x+5y+2z=4
2
3a-1b=7
2a+3b=1
2
5i+4m=1
3i-6m=2
3
4x+4y+1z=24
2x-4y+1z=0
5x-4y-5z=12
2
-1x+1y=2
1x+0y=-3
```

Sample output:

```
x=5.000,y=3.000,z=-2.000
t=-2.286,a=-3.143
x=-4.000,y=0.000,z=-2.000
a=2.000,b=-1.000
i=0.333,m=-0.167
x=4.000,y=2.000,z=0.000
x=-3.000,y=-1.000
```